CLAIMS

What is claimed is:

1. A method, comprising:

receiving a plurality of data packets by a processing system via a network during a pre-boot runtime of the processing system, each of the plurality of data packets containing one of a corresponding plurality of data segments;

parsing the plurality of data packets using a network protocol stack to extract the plurality of data segments during the pre-boot runtime, a portion of the network protocol stack executed in a hardware entity of the processing system; and

transferring the plurality of data segments into system memory of the processing system during the pre-boot runtime.

2. The method of claim 1 wherein transferring the plurality of data segments into the system memory further comprises transferring the plurality of data segments directly into the system memory via a remote direct memory access protocol ("RDMA").

3. The method of claim 1 wherein the portion of the network protocol stack executed in the hardware entity includes a transmission control protocol over Internet protocol ("TCP/IP").

4. The method of claim 3 wherein the TCP/IP is implemented in the hardware entity using a TCP/IP Offload Engine ("TOE").

5. The method of claim 3 wherein the portion of the network protocol stack executed in the hardware entity further includes a user datagram protocol over Internet protocol ("UDP/IP").

6. The method of claim 1, further comprising:

pre-posting a buffer in the system memory of the processing system prior to receiving a first one of the plurality of data segments, the buffer having a size corresponding to a data block, the plurality of data segments comprising segments of the data block, and

wherein transferring the plurality of data segments into the system memory includes transferring the plurality of data segments into the buffer in the system memory.

7. The method of claim 6 wherein the data block comprises a boot agent and a boot image, the boot agent containing instructions for the processing system execute to determine what to do with the boot image.

8. The method of claim 7 wherein the boot image comprises an operating system for executing on the processing system.

9. The method of claim 7, further comprising:

executing the boot agent;

copying the boot image onto a hard disk drive of the processing system;

resetting the processing system; and

booting the processing system from the boot image copied to the hard disk drive.

10. The method of claim 9 wherein copying the boot image onto the hard disk drive includes copying over a previous boot image currently stored on the hard disk drive with the boot image to repurpose the processing system.

11. The method of claim 9 wherein copying the boot image onto the hard disk drive includes copying the boot image onto the hard disk drive having no previous boot image to provision the processing system with the boot image.

12. The method of claim 7, further comprising:

executing the boot agent; and

branching into the boot image from the boot agent to initialize an operating system embedded within the boot image; and

executing the operating system.

13. The method of claim 1 wherein the hardware entity comprises a network interface card.

14. A machine-accessible medium that provides instructions that, if executed by a machine, will cause the machine to perform operations comprising:

receiving a plurality of data packets by a processing system via a network, each of the plurality of data packets containing one of a corresponding plurality of data segments;

parsing the plurality of data packets using a network protocol stack to extract the plurality of data segments during a pre-boot runtime of the processing system, a portion of the network protocol stack executed in a hardware entity of the processing system; and

transferring the plurality of data segments into system memory of the processing system during the pre-boot runtime.

15. The machine-accessible medium of claim 14, further providing instructions that, if executed by the machine, will cause the machine to perform the operations wherein transferring the plurality of data segments into the system memory further comprises transferring the plurality of data segments directly into the system memory via a remote direct memory access protocol ("RDMA").

16. The machine-accessible medium of claim 14, further providing instructions that, if executed by the machine, will cause the machine to perform the operations wherein the portion of the network protocol stack executed in the hardware entity includes a transmission control protocol over Internet protocol ("TCP/IP").

17. The machine-accessible medium of claim 16, further providing instructions that, if executed by the machine, will cause the machine to perform the operations wherein the TCP/IP is implemented in the hardware entity using a TCP/IP Offload Engine ("TOE").

18. The machine-accessible medium of claim 16, further providing instructions that, if executed by the machine, will cause the machine to perform the operations wherein the portion of the network protocol stack executed in the hardware entity further includes a user datagram protocol over Internet protocol ("UDP/IP").

19. The machine-accessible medium of claim 14, further providing instructions that, if executed by the machine, will cause the machine to perform further operations, comprising:

pre-posting a buffer in the system memory of the processing system prior to receiving a first one of the plurality of data segments, the buffer having a size corresponding to a data block, the plurality of data segments comprising segments of the data block, and

wherein transferring the plurality of data segments into the system memory includes transferring the plurality of data segments into the buffer in the system memory.

20. The machine-accessible medium of claim 19, further providing instructions that, if executed by the machine, will cause the machine to perform the operations wherein the data block comprises a boot agent and a boot image, the boot agent containing instructions for the processing system execute to determine what to do with the boot image.

21. The machine-accessible medium of claim 20, further providing instructions that, if executed by the machine, will cause the machine to perform the further operations, comprising:

executing the boot agent;

copying the boot image onto a hard disk drive of the processing system;

resetting the processing system; and

booting the processing system from the boot image copied to the hard disk drive.

22. The machine-accessible medium of claim 20, further providing instructions that, if executed by the machine, will cause the machine to perform the further operations, comprising:

executing the boot agent; and

branching into the boot image from the boot agent to initialize an operating system embedded within the boot image; and

executing the operating system.

23. A processing system, comprising:

a processor to execute an operating system and application software;

system memory communicatively coupled to the processor;

a communication link communicatively coupled to the system memory and to couple to a network, the communication link including a network protocol offload engine to implement a portion of a network protocol stack; and

a flash memory unit communicatively coupled to the processor, the flash memory unit having stored therein a first pre-boot application to request transfer of a first data block from the network into the system memory, the first pre-boot application to interact with the network protocol offload engine to initiate transfer of the first data block during a pre-boot runtime of the processing system.

24. The processing system of claim 23 wherein the network protocol offload engine provides a transmission control protocol/internet protocol ("TCP/IP") service to the first pre-boot application.

25. The processing system of claim 23 wherein the network protocol offload engine provides a remote direct memory access service to transfer the first data block directly into system memory without involving the processor of the processing system.

26. The processing system of claim 25 wherein the first pre-boot application pre-posts a buffer in the system memory having a size corresponding to the data block prior to initiating transfer of the first data block.

27. The processing system of claim 23 wherein the first pre-boot application is an extensible firmware interface driver executed during a pre-boot runtime of the processing system.

28. The processing system of claim 23 wherein the processing system comprises a processing blade to be mounted in a blade server chassis.

29. The processing system of claim 23 wherein the first pre-boot application is to request transfer of the first data block including a boot agent and a boot image, the boot image including an operating system.